Reduced SWAP-C VICTORY Services

Execution and Performance Evaluation

GVSETS

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
03 AUG 2012	**Briefing**	**01-07-2012 to 01-08-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Reduced Swap-C Victory Services	**W56hzv-11-c-l737**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
Mark Russell	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
UBT, Inc.,3250 W Big Beaver Rd, Suite 329,Troy,Mi,48084	**; #23233**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
U.S. Army TARDEC, 6501 East Eleven Mile Rd, Warren, Mi, 48397-5000	**TARDEC**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
	#23233

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Submitted to 2012 NDIA Ground Vehicle Systems Engineering and Technology Symposium August 14-16 Troy, Michigan

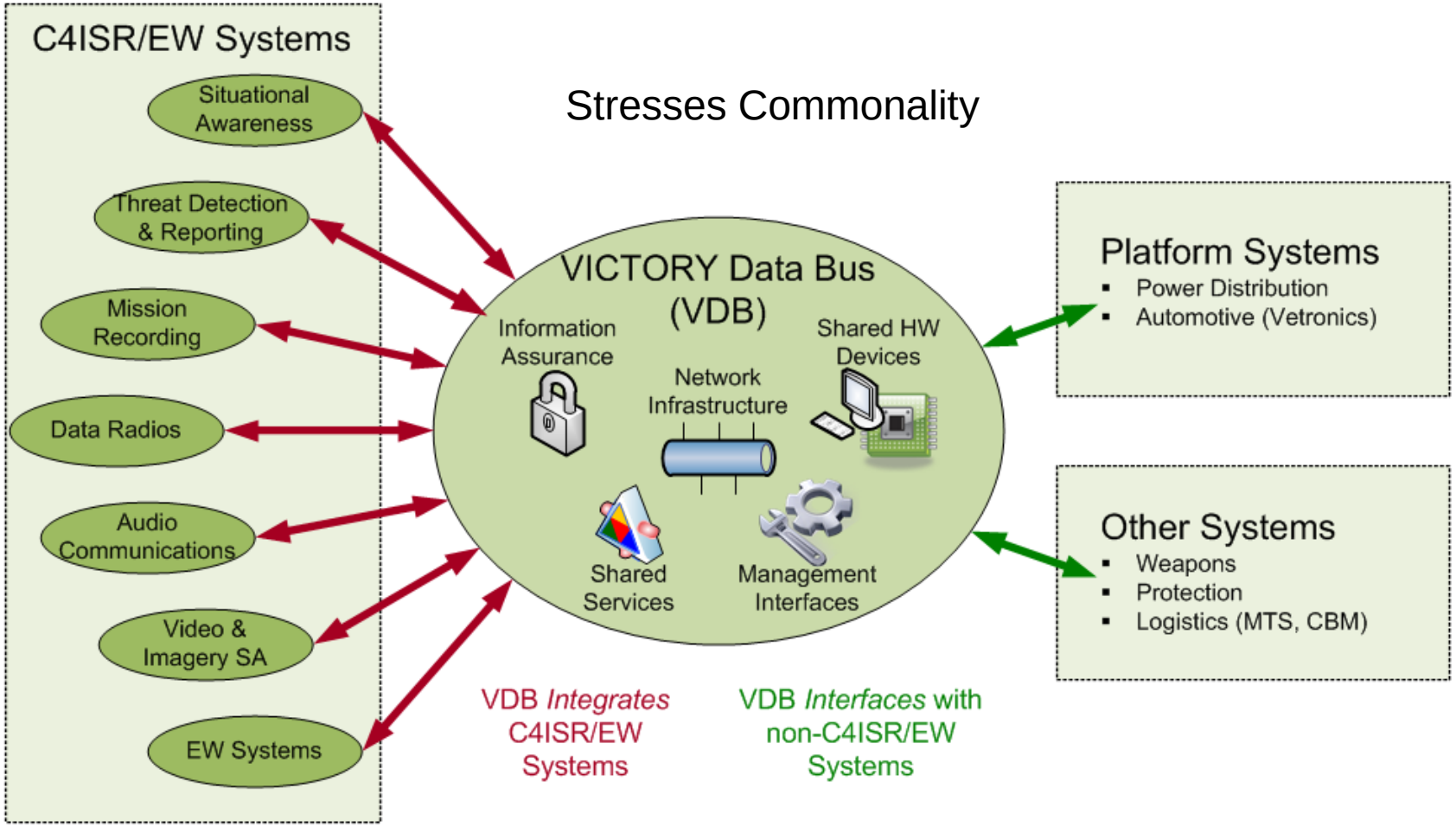
14. ABSTRACT

-Executing multiple VICTORY data services, and reading multiple VICTORY-compliant sensors at the same time resulted in the following performance measurements for the system -0.64 Amps / 3.15 Watts Power Consumption at run-time. -Roughly 0.77% System Memory Utilization per Service 1.05%, 0.433%., 0.531% average CPU utilization for Position, Direction of Travel, and Orientation Services, respectively. -Less than 1.7 milliseconds processing time (Building and Publishing Full Featured VICTORY XML Messages). -A delta of 68.97 Watts between workstation and BeagleBoard, which is 95.6% more efficient. ?Therefore, boards based on this type of architecture are an excellent candidate for running VICTORY services while providing significant reductions in SWAP-C for the VICTORY 1.0 project.

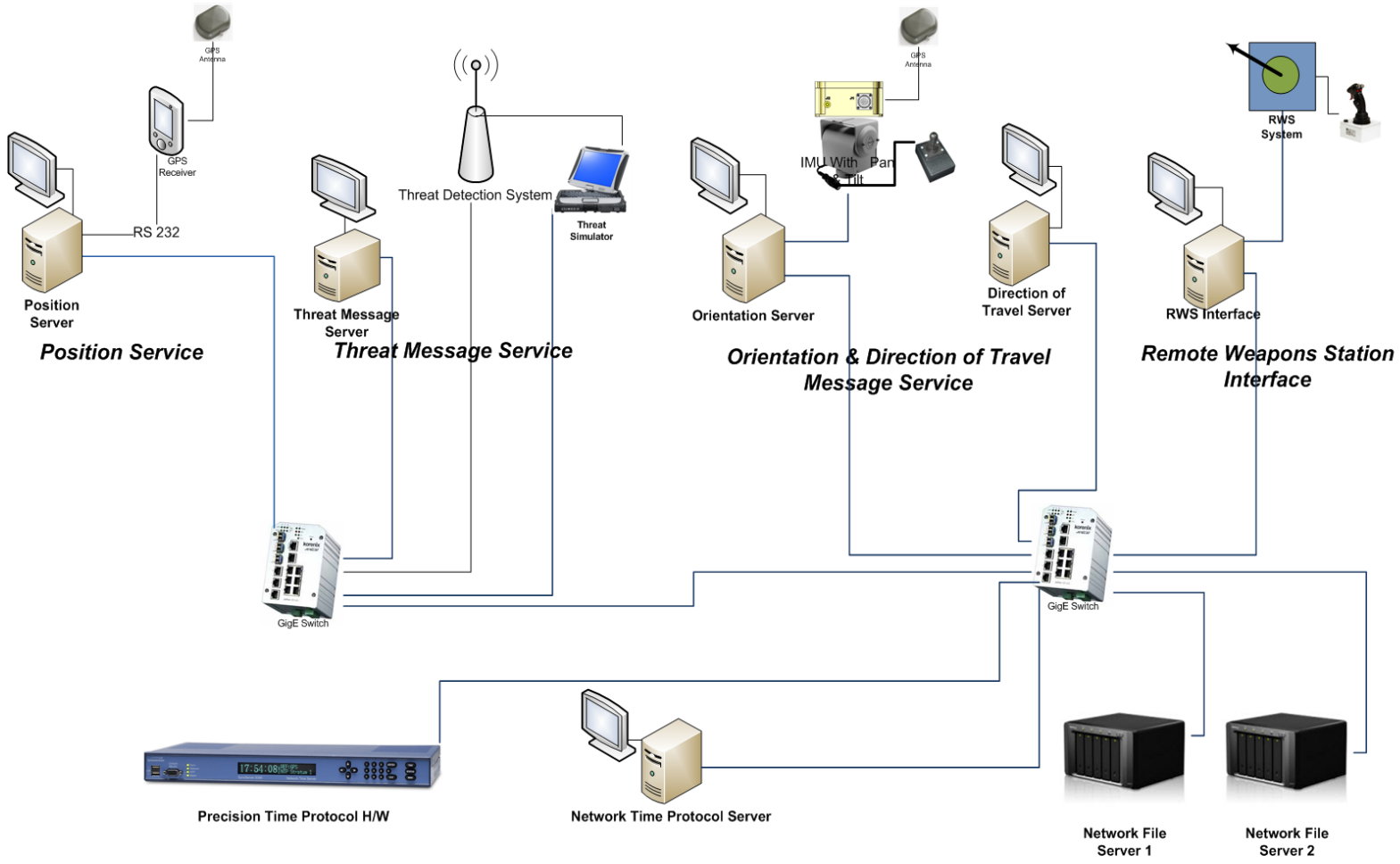
15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Public Release**	**23**	

VICTORY Architecture Overview



VICTORY Services Overview



Small SWAP-C / Embedded System – The "BeagleBoard"

BeagleBoard – xM

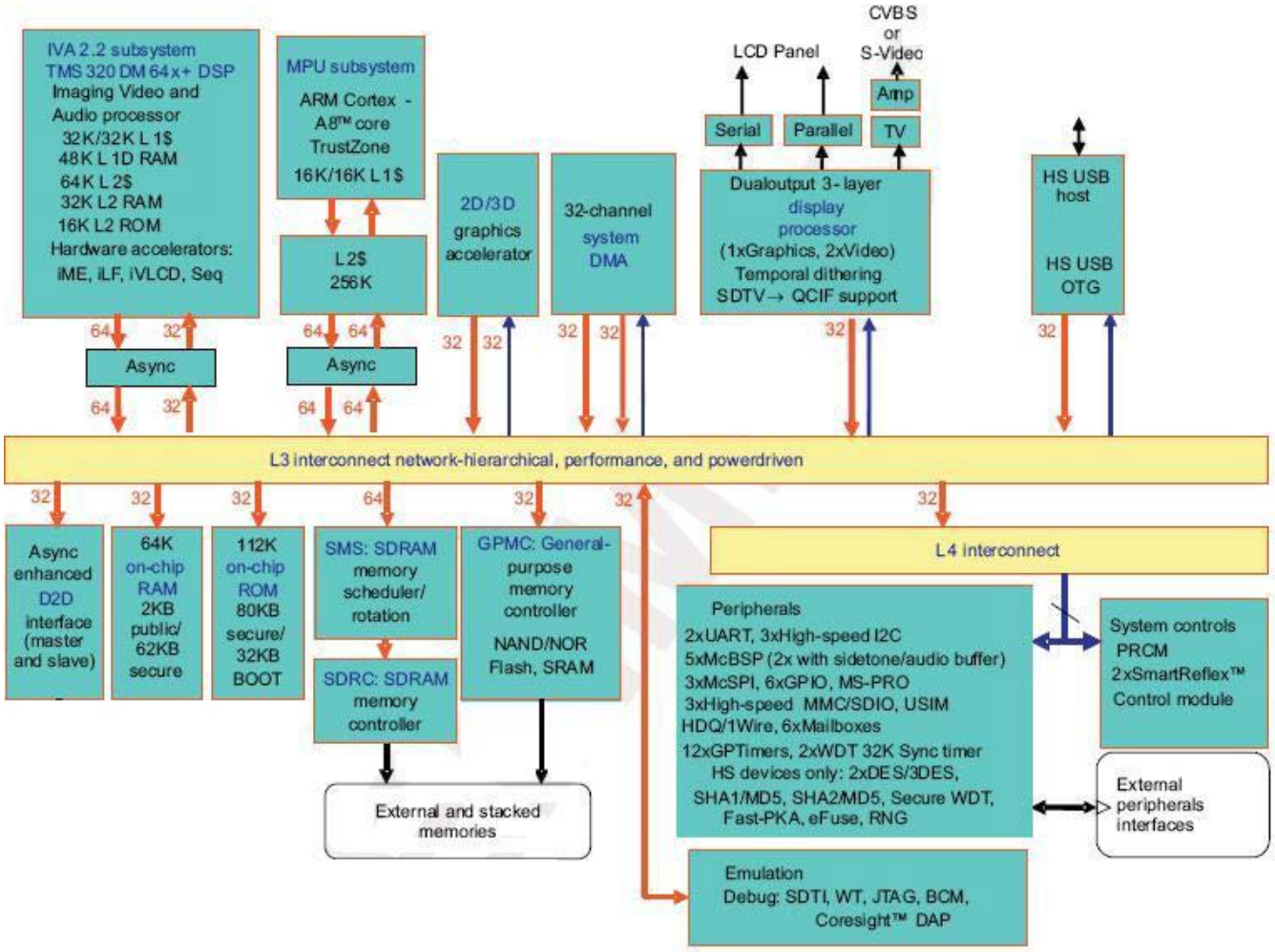
Single Board Computer – Development Board

Hardware:
ARM® Cortex™-A8 CPU
3D graphics accelerator
Multiple external I/O ports

Low Cost:
$150 / board

Low Weight:
Around 1 lb.

High Level Block Diagram DM3730 processor



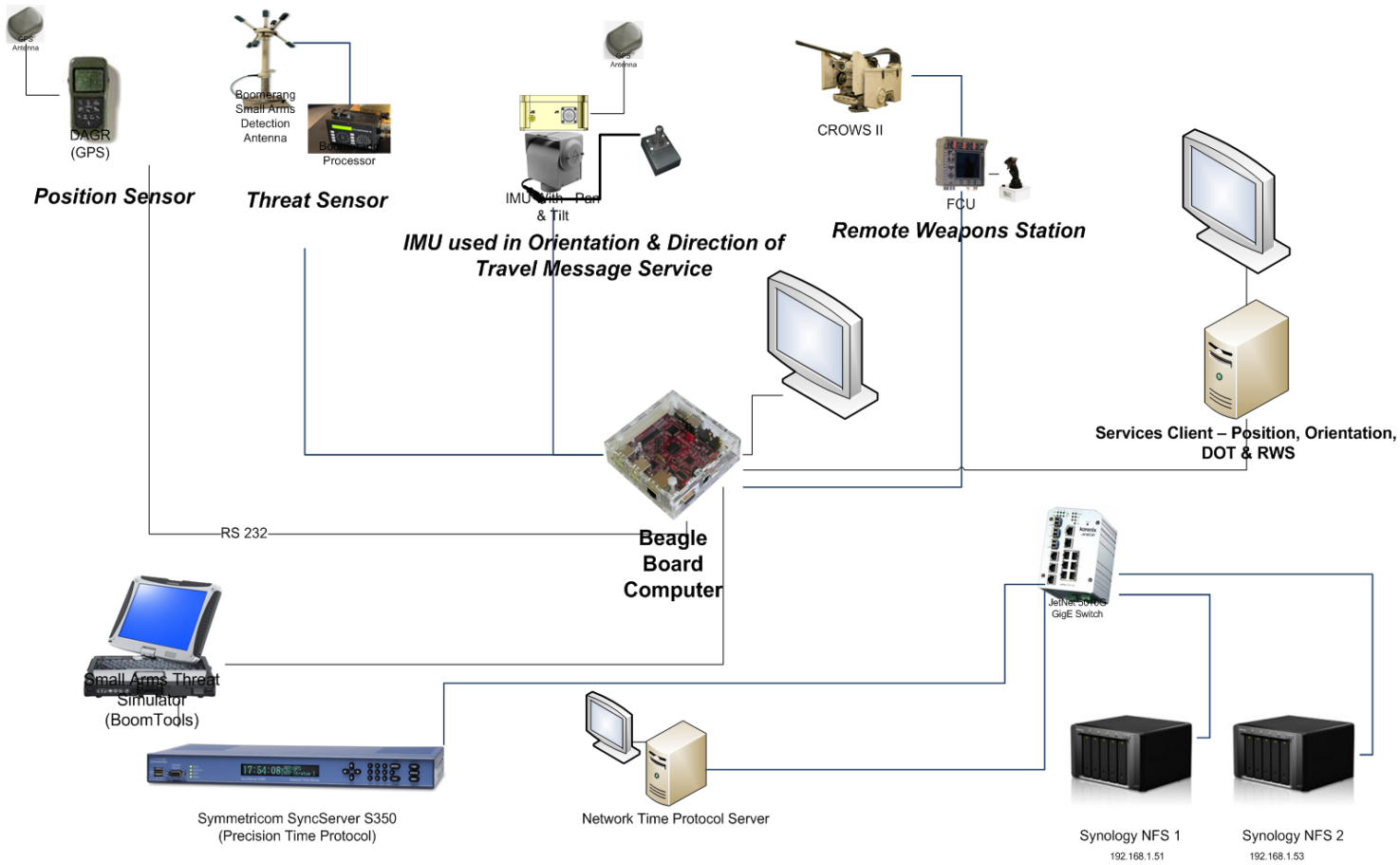
Microprocessor unit (MPU) subsystem based on the ARM Cortex-A8™ microprocessor

- POP Memory interface
- 4Gb MDDR (512Mbytes)
- 24 Bit RGB Display interface (DSS)
- SD/MMC interface
- USB OTG interface
- NTSC/PAL/S-Video output
- Power management
- Serial interface
- I2C interface
- I2S Audio interface (McBSP2)
- Expansion McBSP1
- JTAG debugging interface

Successful Applications of the Beagleboard

- Popular BeagleBoard Projects
  - Automatic dog-walker robot
  - Location aware weather station
  - Android mini-desktop computer
  - Autonomous All Terrain Vehicle
  - 30 fps full screen OpenGL rendering
- The following Operating Systems have been ported to this architecture:
  - Ubuntu
  - Android
  - Angstrom
  - Symbian
  - Gentoo
  - QNX Neutrino

The "After" Picture for VICTORY Services Running in an Embedded Environment



VIDS Services Physical Diagram using Beagle Board Computer

O/S & Software Deployment

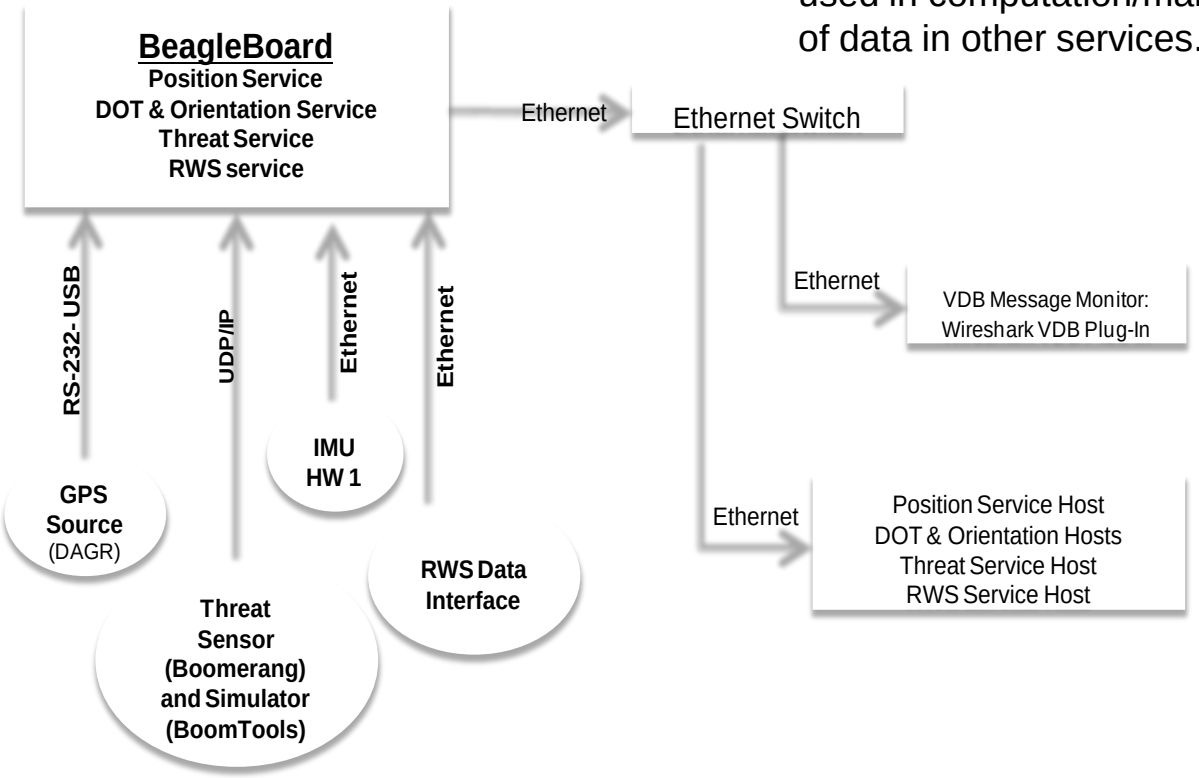
- Finding Suitable Operating System
  - Angstrom – default OS out of the box.
    - Difficulty finding the correct drivers and libraries
  - Ubuntu 10 – quickly switched to this OS as there was much better community support and therefore readily available drivers. (Beagleboard – elinux.org)
    - Difficult to install, given the serial / hyper-terminal interface means of installation. Despite proper connection settings, words and graphics became mangled.
  - Memory card became corrupt, single point of failure. Had to start over from scratch.

- Code-sourcery – open source ARM-GCC compiler
  - Linux development hosts create cross-compiled binaries to run on the target Beagleboard
    - Having issues compiling code against the arm openssl library
  - It is also possible to develop and compile software directly on the hardware itself, although it takes considerable more time.

Physical Interface Diagram for BeagleBoard's Services

- Single point of service for data

Service data consumed. It is displayed/validated, or used in computation/manipulation of data in other services.



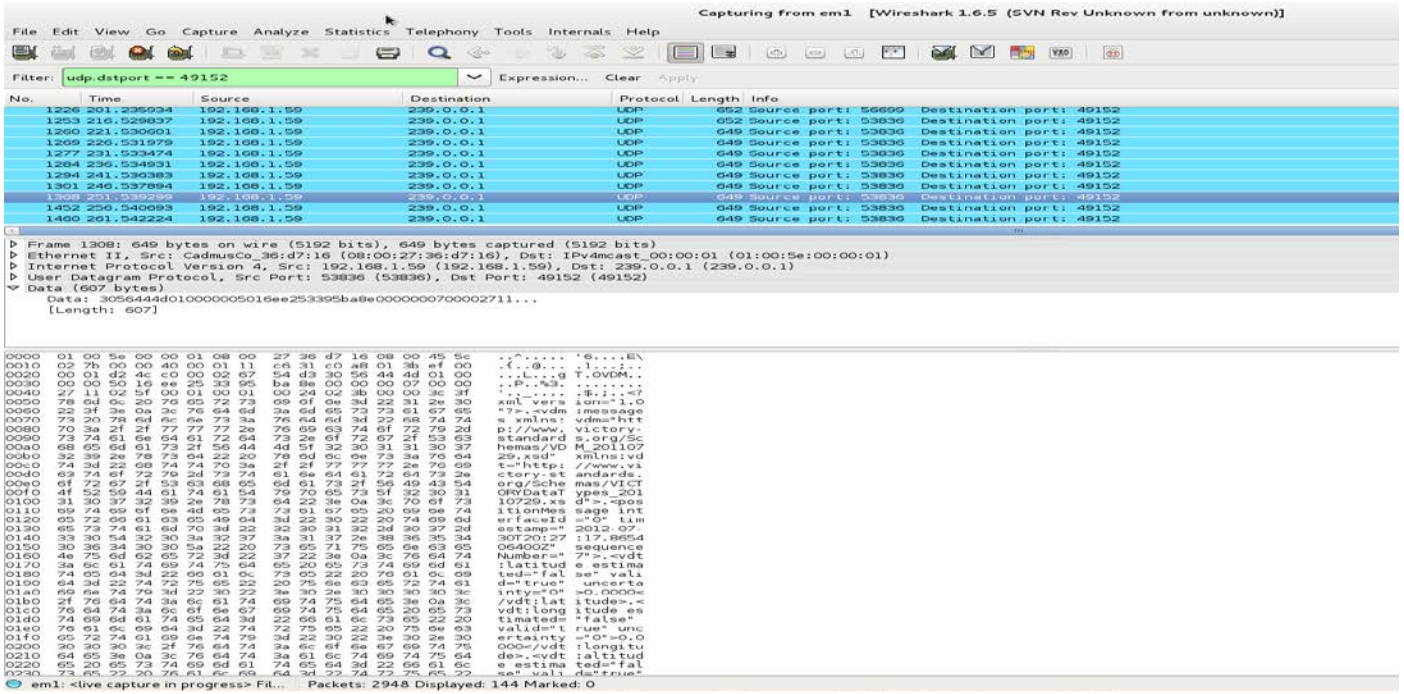
Software Benchmarking

- Wish to determine how efficient our SBC BeagleBoard is at running VICTORY services.
  1. Record statistics for each process' system utilization
  2. Determine the amount of memory consumption
  3. Compare processing time of Victory Data Messages
  4. Determine how much overhead comes from the software versus the external sensors / peripherals
  5. Determine the efficiency at which web-based gSOAP service model executes.

Software Benchmarking

- The Linux kernel records the amount of clock cycles that it performs actual work in 1 second. These units are known as Jiffies.
- Each running process stores metadata about itself in the virtual /proc file system. It is possible to obtain the following:
  - (1) The amount of Jiffies per this process. Using this data versus the entire summation of Jiffies for each process executing, you can obtain the percentage of CPU utilization.
    - Trigger data stores for minimum and maximum percentages
    - Obtain average percentage
  - (2) The amount of memory utilization. After our services are instantiated, they do not malloc any data. The memory usage remains fairly constant.

Sample Position Victory Data Message (VDM)




```
<?xml version="1.0"?>
  <vdm:messages xmlns:vdm=http://www.victory-standards.org/Schemas/VDM.xsd
   xmlns:vdt=http://www.victory-standards.org/Schemas/VICTORYDataTypes.xsd">
    <positionMessage timestamp=" 2012-07-31T15:15:44.5080375397Z" sequenceNumber="1" interfaceID="0">
      <vdt:latitude valid="true" uncertainty="0" estimated="false">38.898648</vdt:latitude>
      <vdt:longitude valid="true" uncertainty="0" estimated="false">-77.037692</ vdt:longitude>
      <vdt:altitude valid="false">0</ vdt:altitude>
      <MGRS> 4QFJ1234567890</MGRS>
    </positionMessage>
</vdm:messages>
</xml>
```

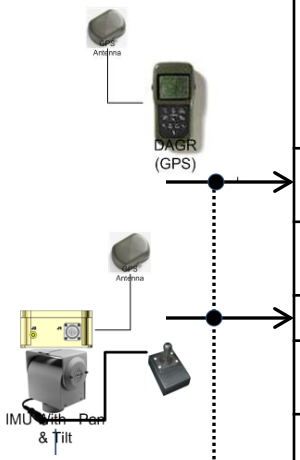
Position Service Benchmarking

- (1) **Each VICTORY process' system and (2) memory utilization**
  - Average CPU Utilization: 1.053%
  - Average Memory Consumption: 0.772%
  - Average System Jiffies: 0.666
  - Minimum CPU Utilization: 0%
  - Maximum CPU Utilization: 3.061%

- (3) **Processing Time of VDMs**
  - XML Build Time Average. BeagleBoard: 713,100 nsec vs. Workstation Build Time: 83,000 nsec
  - VDM Transmit Time Average. BeagleBoard 953,000 nsec vs. Workstation: 100,000 nsec

Additional Overhead on System For Each Sensor Type

Comparison of overhead from VICTORY software versus the external sensors / peripherals



Device	Service	Average CPU Utilization	Average Process Jiffies
GPS Sensor	Position	1.053%	.666
Synthetic GPS Data	Position	.375%	.388
INS Sensor	Orientation	.433%	.665
Synthetic INS Data	Orientation	.395%	.388
Obtained via Position & Orientation Data	Direction of Travel	.531%	.659
Synthetic Direction of Travel Data	Direction of Travel	.404%	.388

When running VICTORY services that consume sensor data, it is found that each sensor accounts for 64.3%, 8.7%, and 23.9% of the process execution, respectively for each service. Running at 4800 Baud, the receive, process, and transmit states for the serial GPS device are the largest bottleneck in terms of sensor/system performance.

GSOAP Service Model Efficiency

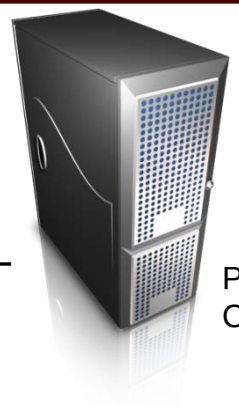
Position Management Server

Get

Position Data

.005 secs

Response

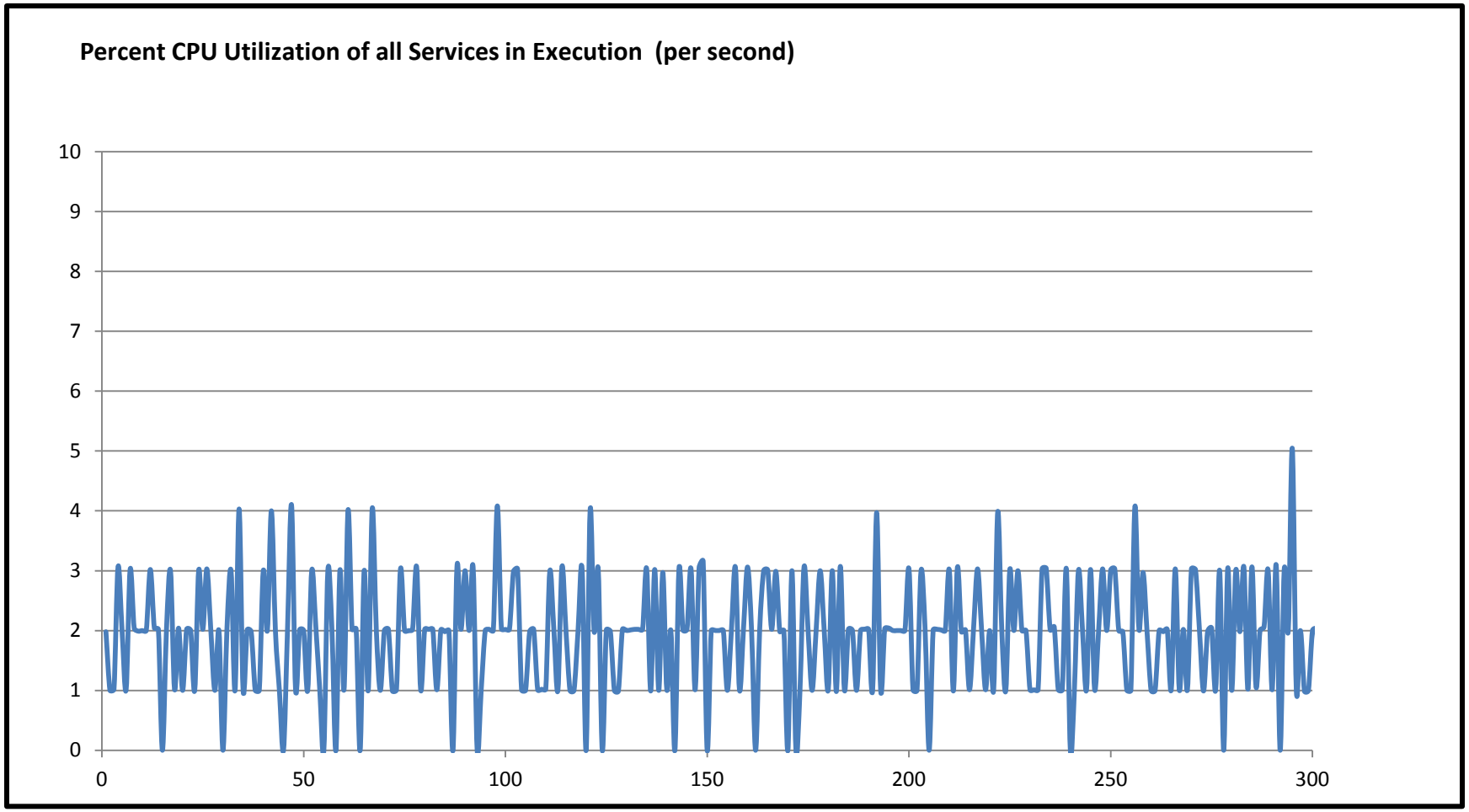


Position Management Client

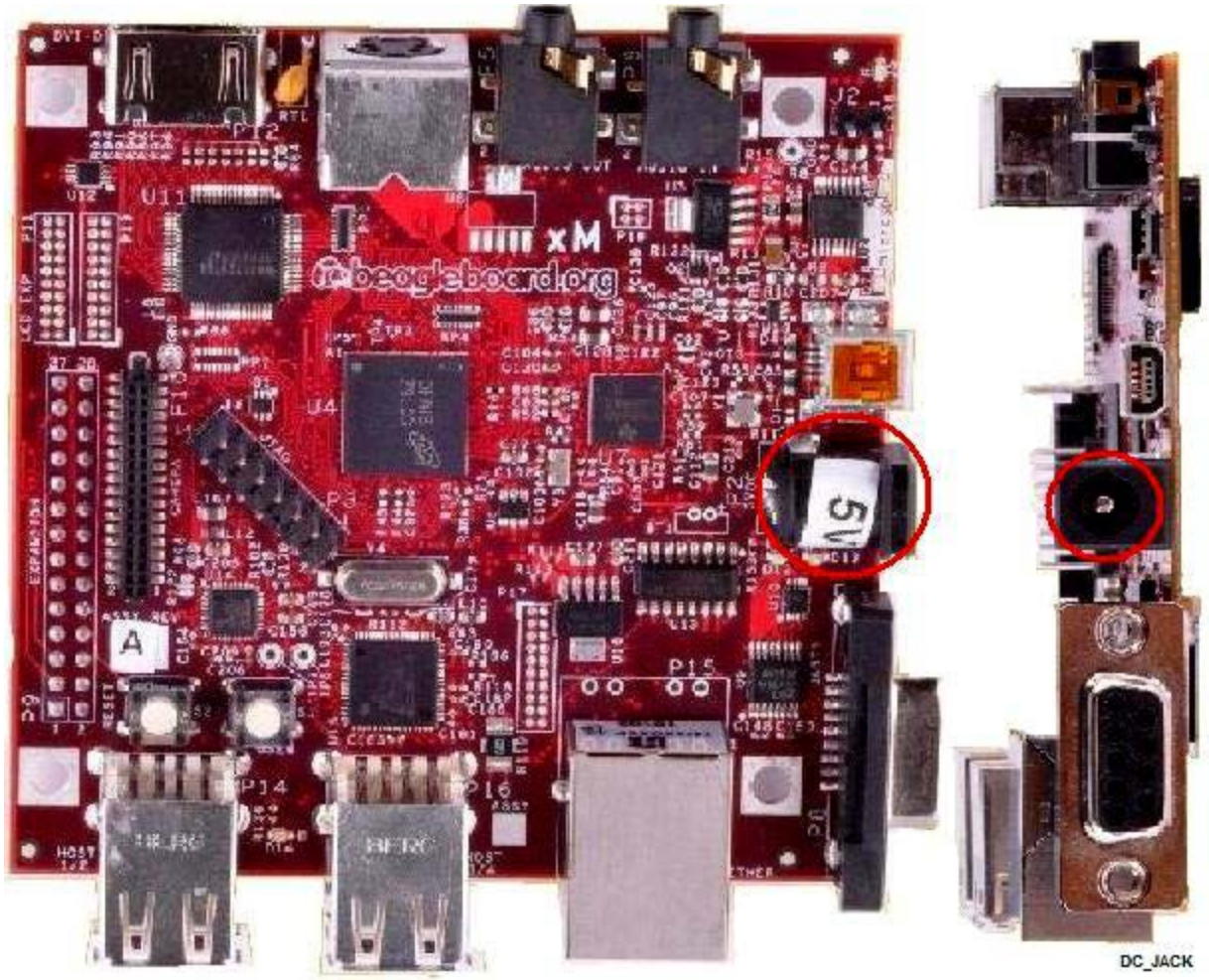
Frequency	% Average CPU Utilization	# of dropped Ethernet packets
1 Hz	.54	0
10 Hz	2.77	0
100 Hz	25.22	0
200 Hz	49.14	0

Aggregation of the Core VICTORY Services in Execution

VEA VEHICLE ELECTRONICS AND ARCHITECTURE

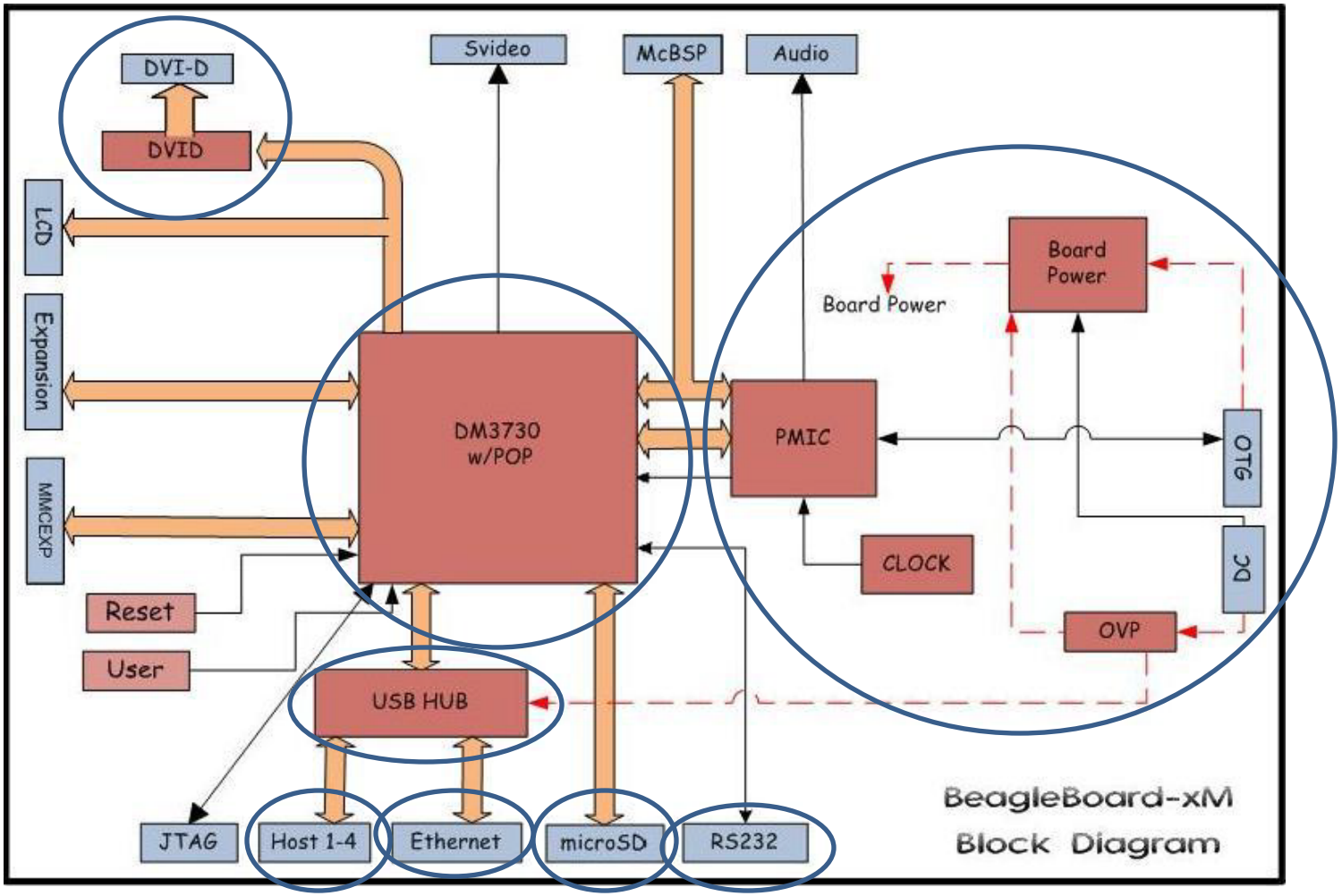


Measuring Power Consumption



- Tektronix DPO 4104 Oscilloscope - used to collect and store data on current draw from current probe and amplifier.
- Tektronix 6302 Current Probe - non-invasive probe for measuring electrical current through a wire.
- We used this hardware to collect the power consumption during system execution.

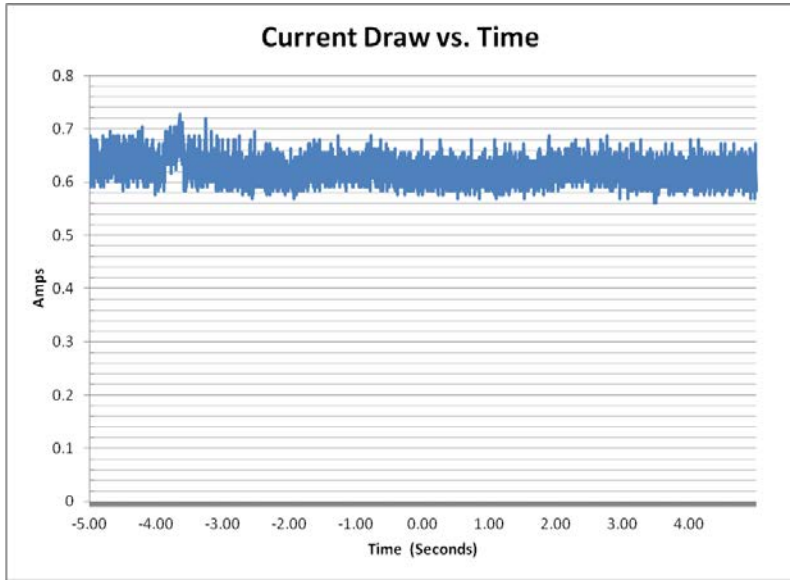
High Level Block Diagram of Components Under Test



Power Draw of BeagleBoard in Operation

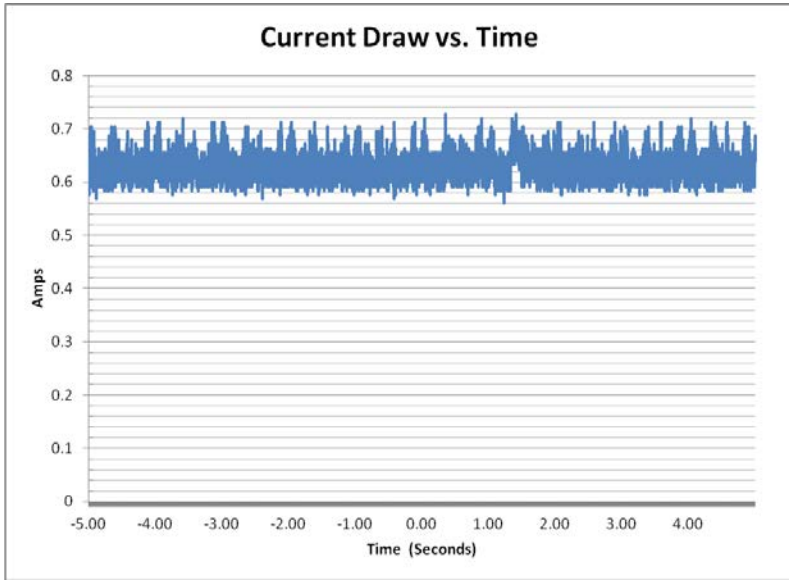
Measurements taken while running Ubuntu OS (executing a 2.6 Linux kernel) and running a stripped-down X-windows GUI that was being driven to an LCD display.

Stable average of 0.62 Amps / 3.10 Watts



Current Draw (Amps) - VICTORY Services at Rest

Stable average of 0.64 Amps / 3.2 Watts



Current Draw (Amps) with VICTORY Services in Operation

Power Draw Comparison

BeagleBoard xM

3.2 Watts

Dell Workstation PCs running 6-cores:

72.12 Watts

95.6% more efficient

The same tests were executed while running the same VICTORY services on Dell Workstation PC; this resulted in an observed 0.85 Amps RMS. This consequently computes to a total average power consumption of 72.12 Watts. Thus, the magnitude of 72.12 Watts versus the 3.15 Watts measured on the BeagleBoard represents a savings of 68.97 Watts, and is 95.6% more efficiently to run the very same services.

Summary

- Executing multiple VICTORY data services, and reading multiple VICTORY-compliant sensors at the same time resulted in the following performance measurements for the system:
  - 0.64 Amps / 3.15 Watts Power Consumption at run-time.
  - Roughly 0.77% System Memory Utilization per Service
  - 1.05%, 0.433%., 0.531% average CPU utilization for Position, Direction of Travel, and Orientation Services, respectively.
  - Less than 1.7 milliseconds processing time (Building and Publishing Full Featured VICTORY XML Messages).
  - A delta of 68.97 Watts between workstation and BeagleBoard, which is 95.6% more efficient.
- Therefore, boards based on this type of architecture are an excellent candidate for running VICTORY services while providing significant reductions in SWAP-C for the VICTORY 1.0 project.

Final Thoughts

VEA
VEHICLE ELECTRONICS AND ARCHITECTURE

? Questions ?

Disclaimer: Reference herein to any specific commercial company, product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the Department of the Army (DoA). The opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or the DoA, and shall not be used for advertising or product endorsement purposes.

Disclaimer: Reference herein to any specific commercial company, product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the Department of the Army (DoA). The opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or the DoA, and for advertising or product endorsement purposes.shall not be used